REMARKS

This Amendment is in response to the Office Action dated August 24, 2004. In the Office Action, claims 2-6, 8 and 10 were objected to, claims 1-10 were rejected under 35 USC §112 and claims 1-10 were rejected under 35 USC §103. By this Amendment, the drawings, specification and claims 1-10 are amended. Currently pending claims 1-10 are believed allowable, with claims 1, 6 and 10 being independent claims.

PRIORITY:

The Office Action states that a certified copy of Priority Document 00128489.2 has not been filed for the present Application. Office Action, paragraph 3. The Applicant respectfully submits that a certified copy of Priority Document 00128489.2 was submitted on February 19, 2002, as evidenced by a copy of the stamped return postcard provided herewith, and as stated in paragraph 2 of the Office Action.

AMENDMENTS TO THE DRAWINGS:

The Office Action indicates that Figs. 1-6, 9 and 11 should be labeled with the legend "Prior Art". Office Action, paragraph 4. By this Amendment, Figs. 1-5 and 9 are amended to include a "Prior Art" legend. It is respectfully submitted, however, that Figs. 6 and 11 do not to illustrate prior art and that the specification does not indicate that these figures illustrate prior art. Accordingly, Figs. 6 and 11 are not amended with a Prior Art legend.

AMENDMENTS TO THE SPECIFICATION:

The Office Action indicates that the specification contains numerous grammatical errors. Office Action, paragraph 7. By this Amendment, such grammatical and typographical errors are corrected. No new matter is believed to be introduced to the Application by these amendments to the specification.

CLAIM OBJECTIONS:

Claim 2:

Claim 2 was objected to because of the term "dependency-signal" in the last line. Office Action, paragraph 9b. By this Amendment, claim 2 is amended to recite a "dependency signal" (hyphen removed), as suggested by the Examiner.

Claim 3:

Claim 3 was objected to for the use of the term "no-dependency-signal". Office Action, paragraph 9c. By this Amendment, claim 3 is amended to recite a "no-dependency signal" (hyphen after "dependency" removed), as suggested by the Examiner.

Claim 4:

Claim 4 was objected to because of the use of the term "step" rather than "steps". Office Action, paragraph 9d. By this Amendment, the term "step" in claim 4 is amended to "steps", as suggested by the Examiner.

Claim 4 was also objected to for the use of the term "committed-status-flag". Office Action, paragraph 9e. By this Amendment, claim 4 is amended to recite "committed-status flag" (hyphen after "status" removed), as suggested by the Examiner.

Claim 4 was further objected to because of the term "dependencyfor". Office Action, paragraph 9f. The Applicant respectfully submits that this term does not appear in claim 4.

Claim 5:

Claim 5 was objected to for reciting "step" rather than "steps". Office Action, paragraph 9g. By this Amendment, the term "step" in claim 5 is amended to "steps", as suggested by the Examiner.

Claim 5 was also objected to for the use of the term "committed-status-flag". Office Action, paragraph 9h. By this Amendment, claim 5 is amended to recite "committed-status flag" (hyphen after "status" removed), as suggested by the Examiner.

Claim 6:

Claim 6 was objected to for the use of the term "no-dependency-signal". Office Action, paragraph 9i. By this Amendment, claim 6 is amended to recite "no-dependency signal" (hyphen after "dependency" removed), as suggested by the Examiner.

Claim 8:

Claim 8 was objected to for reciting "valid-bits" rather than "valid bits". Office Action, paragraph 9j. By this Amendment, the term "valid-bits" in claim 8 is amended to "valid bits", as suggested by the Examiner.

Claim 10:

Claim 10 was objected for reciting "no-dependency-signal". Office Action, paragraph 9k. By this Amendment, claim 10 is amended to recite "no dependency signal" (hyphen after "dependency" removed), as suggested by the Examiner.

CLAIM REJECTIONS UNDER 35 USC §112:

Claims 6-10:

Claims 6-10 were rejected under 35 USC §112 as reciting subject matter not described in the specification. Office Action, paragraph 11. By this Amendment, the specification is amended at paragraph [0036] to disclose that operations of the invention may be implemented in either hardware or software.

Claim 1:

Claim 1 was rejected under 35 USC §112 as containing indefinite language in the preamble. Office Action, paragraph 15. By this Amendment, the preamble is amended to more clearly recite an instruction pipeline for an out-of-order processor. No new matter is introduced by this rewording of the claim and the claim amendment is not made to overcome the cited art.

Claim 1 also rejected under 35 USC §112 as being indefinite as to which of the "logic target address" or the "one or more instructions" are stored in the temporary buffer. Office Action, paragraph 16. Claim 1 is amended herein to more clearly recite that the "one or more instructions" are stored in the temporary buffer. No new matter is introduced by this rewording of the claim and the claim amendment is not made to overcome the cited art.

Claim 2:

Claim 2 was rejected under 35 USC §112 as being indefinite for the phrase "in case of a match". Office Action, paragraph 17. By this Amendment, the phrase "in case the no-dependency signal is not active" is substituted for the phrase "in case of a match" in order to more clearly define the metes and bounds of the invention claimed. This amendment is not made to overcome the cited art and does not introduce new matter to the Application.

Claim 3:

Claim 3 was rejected under 35 USC §112 as being indefinite for reciting that registers are evaluated "into" a signal generation. Office Action, paragraph 18. By this Amendment, claim 3 is modified to recite that the registers are evaluated to generate the no-dependency signal. No new matter is introduced by this rewording of the claim. This claim amendment is not made to overcome the cited art.

Claim 3 was also rejected under 35 USC §112 as being indefinite for reciting the term "non-architected target registers". Office Action, paragraph 19. By this amendment, "speculative target registers" is substituted for this term to more clearly define the metes and bounds of the invention claimed. Support for this amendment can be found at least at paragraph [0040] of the Application. As such, no new matter is introduced by this claim amendment. Furthermore, amendment to claim 3 is not made to overcome the cited art.

Claim 4:

Claim 4 was rejected under 35 USC §112 as being indefinite for the phrase "in case of a match". Office Action, paragraph 20. By this Amendment, the phrase "in case the no-dependency signal is not active" is substituted for the phrase "in case of a match" in order to more clearly define the metes and bounds of the invention claimed. This amendment is not made to overcome the cited art and does not introduce new matter to the Application.

Claim 5:

Claim 5 was rejected under 35 USC §112 as being indefinite for the phrase "in case of a match". Office Action, paragraph 21. By this Amendment, the phrase "in case the no-dependency signal is not active" is substituted for the phrase "in case of a match" in order to more clearly define the metes and bounds of the invention claimed. This amendment is not made to overcome the cited art and does not introduce new matter to the Application.

Claim 6:

Claim 6 was rejected under 35 USC §112 as being indefinite for the phrase "post-connected". Office Action, paragraph 22. It is respectfully submitted that the phrase "post-connected" does not appear in claim 6.

Nevertheless, claims 7 and 9 are amended to replace recitation of the phrase "post-connected" with the phrase "coupled with". These amendments are not made to overcome the cited art and do not introduce new matter to the Application.

Claim 8:

Claim 8 was rejected under 35 USC §112 for lack of antecedent basis for the term "target register". Office Action, paragraph 23. By this amendment, claim 8 is modified to provide antecedent basis for this term. This amendment is not made to overcome the cited art.

Claim 9:

Claim 9 was rejected under 35 USC §112 as being indefinite due to the term "selected". Office Action, paragraph 24. By this Amendment, claim 9 is modified to recite that a target register 'valid bits' signal is ANDed in the logic. See Application, paragraph 68 and Fig. 8. No new matter is introduced by this rewording of the claim. Furthermore, this claim amendment is not made to overcome the cited art.

CLAIM REJECTIONS UNDER 35 USC §103:

Claims 1-10 were rejected under 35 USC §103 as being obvious over U.S. Patent No. 5,471,626 to Carnevale et al. ("Carnevale") in view of U.S. Patent No. 5,974,526 to Garg et al. ("Garg").

A *prima facie* case for obviousness can only be made if the combined reference documents teach or suggest all the claim limitations. MPEP 2143.

Claim 1:

Claim 1, as currently amended, recites, "if the no dependency signal is not active, assigning an entry in the temporary buffer to the logic source address of said current instruction; and if the no dependency signal is active, issuing the instruction operand data to an instruction execution unit without assigning the entry in the temporary buffer to the logic source address of said current instruction." Application, claim 1, lines 16-21. Support for this amendment can be found at least at paragraphs 48 and 49 of the present Application.

It is respectfully submitted that the cited documents do not teach or suggest the above-limitations. Specifically, neither Carnevale nor Garg, either alone or in combination, teach issuing an instruction operand data to

an instruction execution unit without assigning an entry in the temporary buffer to the logic source address of the current instruction. For at least this reason, claim 1 is believed allowable and indication of such allowance is earnestly requested.

Claims 2-5:

Claims 2-5 are dependent on and further limit claim 1. Since claim 1 is believed allowable, claims 2-5 are also believed allowable for at least the same reasons as claim 1.

Claim 6:

Claim 6, as currently amended, recites, "a third computer readable code for assigning an entry in the temporary buffer to the logic source address of said current instruction if the no-dependency signal is not active; and a fourth computer readable code for issuing the instruction operand data to an instruction execution unit without assigning the entry in the temporary buffer to the logic source address of said current instruction if the no-dependency signal is active." Application, claim 6, lines 12-20. Support for this amendment can be found at least at paragraphs 48 and 49 of the present Application.

It is respectfully submitted that the cited documents do not teach or suggest the above-limitations. Specifically, neither Carnevale nor Garg, either alone or in combination, teach issuing an instruction operand data to an instruction execution unit without assigning an entry in the temporary buffer to the logic source address of the current instruction. For at least this reason, claim 6 is believed allowable and indication of such allowance is earnestly requested.

Claims 7-9:

Claims 7-9 are dependent on and further limit claim 6. Since claim 6 is believed allowable, claims 7-9 are also believed allowable for at least the same reasons as claim 6.

Claim 10:

Claim 10, as currently amended, recites, "a third computer readable code for assigning an entry in the temporary buffer to the logic source address of said current instruction if the no-dependency signal is not active; and a fourth computer readable code for issuing the instruction operand data to an instruction execution unit without assigning the entry in

the temporary buffer to the logic source address of said current instruction if the no-dependency signal is active." Application, claim 10, lines 12-20. Support for this amendment can be found at least at paragraphs 48 and 49 of the present Application.

It is respectfully submitted that the cited documents do not teach or suggest the above-limitations. Specifically, neither Carnevale nor Garg, either alone or in combination, teach issuing an instruction operand data to an instruction execution unit without assigning an entry in the temporary buffer to the logic source address of the current instruction. For at least this reason, claim 10 is believed allowable and indication of such allowance is earnestly requested.

CONCLUSION

In view of the forgoing remarks, it is respectfully submitted that this case is now in condition for allowance and such action is respectfully requested. If any points remain at issue that the Examiner feels could best be resolved by a telephone interview, the Examiner is urged to contact the attorney below.

No fee is believed due with this Amendment, however, should a fee be required please charge Deposit Account 50-0510. Should any extensions of time be required, please consider this a petition thereof and charge Deposit Account 50-0510 the required fee.

Respectfully submitted,

Dated: November 24, 2004

Ido Tuchman, Reg. No. 45,924
Law Office of Ido Tuchman
69-60 108th Street, Suite 503
Forest Hills, NY 11375
Telephone (718) 544-1110
Facsimile (718) 544-8588

MARKUP SHEET

[0008] After the instructions have been fetched by a fetch unit, passed through a decode and branch prediction unit, stored in the instruction queue and have been renamed in a renaming unit they are stored ~~ininto~~ in a part of the IWB called reservation station. From the reservation station the instructions may be issued out to a plurality of instruction execution units abbreviated herein as IEU, and the speculative results are stored in a temporary register buffer, called reorder buffer, abbreviated herein as ROB. These speculative results are committed (or retired) in the actual program order thereby transforming the speculative result into the architectural state within a register file, a so-called Architected Register Array, further abbreviated herein as ARA. In this way it is assured that the ~~outprocessor~~ out-of-order processor (also referred to herein as an outprocessor) with respect to its architectural state behaves like an in-order processor.

[0018] It is thus an objective of the present invention to reduce the pipeline length in ~~performance-ertical~~ performance-critical cases.

[0038] It is contemplated that operations described herein can be implemented in either hardware or software. In this basic approach at least the instructions with all source data in an architected state (data resides in the register file) are covered by the dependency check.

[0039] Further, when the step of generating a ~~no "dependency~~ "no dependency" signal comprises the step of comparing a plurality of logic target register addresses and the logic source register address of the current instruction, in case of a match, and the step of generating a dependency for the respective source register (and thereby the instruction becomes dependent on another older instruction), the simplest way to determine ~~nois~~ "no dependency" is disclosed because this corresponds straight-forward to the definition of dependency.

[0040] Further, when "valid"-bits of ~~nontarget~~ non-target registers stored in a storage as e.g., the reorder buffer, which is associated with speculatively calculated instruction result data are involved into the ~~nogeneration~~ no generation, then the advantage is that ~~an~~ additional information is made available saying if a respective target register data stored in the reorder buffer is valid or not. Thus, the no-dependency signal generation covers more cases, i.e., the speculatively calculated cases too, i.e., the method is more effective.

[0041] Further, the ~~inventional~~ concepts of the present invention can be applied as well in a ~~mappingrenaming~~ mapping-renaming scheme. Then a mapping table entry is addressed with a logical source register address of the current instruction, whereby the mapped physical target register address is determined, then, a committed-status-flag in said entry is read, and thus, it is known where the data resides, in the ROB (non-committed), or already in the ARA (committed), then, the logic target register address and the logic source register address of the current instruction are compared, and in case of a match, a dependency-signal is generated for the respective source register.

[0042] In case of a content-addressable memory ~~(CAM)renaming~~ (CAM) renaming scheme, according to a preferred embodiment, the means for determining the dependency of a current instruction comprises a compare logic in which all instructions to be checked for dependency are involved, and a post-connected OR gate.

[0048] The fetch unit dispatches up to 4 instructions each cycle to the IWB in program order. The IWB pipeline starts with renaming--510--the up to 4 dispatched instructions. The renaming process, compares the source registers with the target registers of previous instruction and in case of a match, i.e., a dependency is found, then, the ROB entry of the target is assigned to the source register. Furthermore, ~~a~~ new ROB entries ~~is~~ are allocated for the target register of the instruction. In this ROB entry the speculative results will be stored after execution.

[0054] For the case that there is no dependency (RSEL (0 . . . 63)="00 . . . 00") the "read_ARA" is switched ON by the ROB causing

~~that~~ the operand data ~~will~~ to be read from the ARA (addressed by the logical address). This ends the "read ROB" cycle.

[0062] As ~~became~~ may be apparent from the above illustration and description the present invention discloses the introduction of a "no_dependency" signal that directly tells the select logic 418 that the renamed logic will have all its source data available. In this way, the rename/issue part of the pipeline is reduced to three stages as shown in FIG. 6 where the same reference signs apply.

[0078] Further variations are possible such as for example the used of a mapping table based renaming scheme as discussed in the previous section, and a single register file ~~instead~~ in which an instruction is committed by setting a commit bit for the register file entry (rather then copying the data from the ROB to the ARA data file). This however does not modify the objective of the invention by reducing the pipeline length by the generation of a "no_dependency" signal for the cases were the source data is directly available for the instruction. In this latter case, the validity of the data can be derived from including the valid bit into the ~~negeneration~~ "no dependency" generation.

[ABST] A method and system for operating a high frequency ~~outprocessor~~ out-of-order processor with increased pipeline length. A new scheme is disclosed to reduce the pipeline by the detection and exploitation of so called ~~"no_dependency"~~ "no dependency" for an instruction. A "no dependency" signal tells that all required source data is available for the instruction at least one cycle before the source data valid bit(s) are inserted into the issue queue. Therefore, one or more stages of the pipeline are bypassed. ~~Bypassing the pipeline stages for this "no dependency" conditions is especially important since a no dependency is found when the queue is empty. Furthermore, this bypass is very effective when the queue is relatively empty. Therefore, introducing such a bypass reduces effectively the performance drawback of a longer pipeline.~~